This CVPR 2021 workshop paper is the Open Access version, provided by the Computer Vision Foundation.
Except for this watermark, it is identical to the accepted version;
the final published version of the proceedings is available on IEEE Xplore.

Dissecting the High-Frequency Bias in Convolutional Neural Networks

Antonio A. Abello, Roberto Hirata Jr.
University of São Paulo
Butanta, São Paulo - State of São Paulo, Brazil
abello@ime.usp.br, hirata@ime.usp.br

Zhangyang Wang
The University of Texas at Austin
Austin, TX 78712, United States
atlaswang@utexas.edu

Abstract

For convolutional neural networks (CNNs), a common hypothesis that explains both their generalization capability and their characteristic brittleness is that these models are implicitly regularized to rely on imperceptible high-frequency patterns, more than humans would do. This hypothesis has seen some empirical validation, but most works do not rigorously divide the image frequency spectrum. We present a model to divide the spectrum in disjointed discs based on the distribution of energy and apply simple feature importance procedures to test whether high-frequencies are more important than lower ones. We find evidence that mid or high-level frequencies are disproportionately important for CNNs. The evidence is robust across different datasets and networks. Moreover, we find the diverse effects of the network's attributes, such as architecture and depth, on frequency bias and robustness in general. Code for reproducing our experiments is available at: ***https://github.com/Abello966/FrequencyBiasExperiments***

1. Introduction

The machine learning community dedicates a considerable amount of research to understand Deep Learning's functioning in general and Convolutional Neural Networks (CNNs) in particular. Among various questions, two seem to be most intriguing: first, CNNs are capable of generalization even when they are greatly overparametrized [26]. Second, they seem to be excessively brittle and susceptible to adversarial examples [10]: small, imperceptible perturbations that make a model act in undesirable ways. A common hypothesis that would explain and unite both phenomena is that CNNs are somehow biased towards the higher frequency modes of images. Thus the network would be implicitly regularized to lean on imperceptible yet highly generalizable high-frequency patterns [17, 9]. In turn, this would make a network somehow fragile to noise and other common image corruptions that target especially this region of the frequency spectrum [25]. Additionally, this would make a CNN prone to adversarial examples that exploit how perceptually small changes in images could destroy these patterns [14, 24]. Confirming the existence of this high-frequency bias and understanding its nature would be an important step towards understanding how CNN's work and how to make them more robust.

The cited papers collectively present a reliable amount of evidence for the existence of some high-frequency bias in modern CNNs. However, most of the experiments are based on an intuitive but not rigorous definition of what constitutes a high or low image frequency mode, and some do not account for the fact that information may not be evenly distributed across the spectrum. The conditions and scenarios tested vary on each paper, leading to interesting discussions and conclusions that are also worth aggregating and consolidating on a systematic study. We propose to study the high-frequency bias by separating the image frequency spectrum in bands with the same amount of information. We then use a simple method reminiscent of feature importance procedures in traditional Machine Learning to quantify how much different models, under different circumstances, are biased towards each frequency band.

The rest of this paper is structured as follows: we perform a literature review on this topic in Section 2. Section 3 defines the notation used in this work, presents a quick recapitulation of feature importance metrics and describes our proposed method for separating the frequency spectrum. Next, in Section 4, we describe the experimental scenarios we would like to investigate in this work. After that, in Section 5, we present our results and a discussion of them and con-

clude with final remarks and future research directions on section 6.

2. Related Work

To the best of our knowledge, Jo and Bengio [17] were the first to show that the present generation of neural networks are biased towards higher frequencies, which they called surface statistical regularities. They showed that while a network trained on a low-pass filtered version of the dataset could generalize well to the unfiltered version, a network trained on the original dataset would perform much worse when the test set was low-pass filtered. This generalization gap showed that, while not indispensable, networks would latch on to these high-frequency patterns. They heuristically defined a threshold value for high and another for low frequency manually, adjusting the threshold for each dataset and maintaining the filtered images' human perceptual similarity.

This research led to exciting developments expanding the meaning of surface regularities to that of texture ones. Geirhos et al. [9] went further to propose the *texture hypothesis*, according to which the CNNs are biased more towards textural information than shape. They demonstrate the fact by creating experiments in which the shape and texture information are contradictory, finding out that CNNs tend to consider the texture information more than the shape one.

The high-frequency bias was also approached from the point of view of model robustness and adversarial perturbations. Tsuzuku et al. [23] presented sound theoretical reasons for CNN's sensibility to noise in the format of Fourier basis directions. Searching for directions that were effective in fooling classifiers, they found out that networks had increased sensitivity in some regions of the Fourier spectrum, more critically in what one could call a "middle" to "low" region. Wang et al. [24] proposed that highly generalizable but brittle high-frequency patterns in data may account both for CNN's capacity of generalization and sensitivity to adversarial attacks. They collect image examples where the absence of some higher frequencies, albeit unnoticeable by humans, would fool a CNN. They also performed experiments that associate the images' higher frequency components to memorization and overfitting.

Yin et al. [25] presents another related work that does not deal explicitly with high-frequency bias but shows that high-frequency information can be sufficient for reasonable classifying success if one trains a classifier exclusively on them. Similarly, Brendel et al. [3] achieved a competitive performance using CNNs with limited receptive field size, showing that shape information is not necessary, and texture information may be sufficient for image classification. [19] has provided a visualization study of CNN sensitivity to translations.

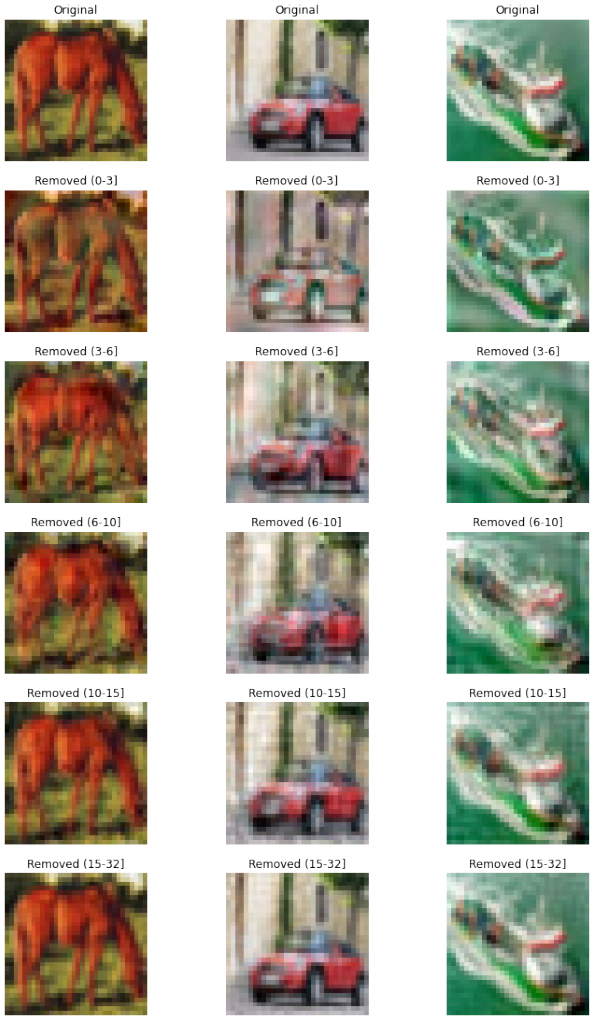

Figure 1. Example of distorted CIFAR10 images according to our model. Notice how color and edges are mixed in the first few bands, but the effect is barely noticeable in the last two

3. Method

We represent an image as a matrix, X, of pixel intensities, i.e., $X \in R^{N \times M}$, $X[p, q] \in R, p \in [0, N-1], q \in [0, M-1]$. We will omit channel information for simplicity, but all image operations are assumed to be applied channel-wise when relevant. When we refer to the Fourier transform of an image and its inverse, we are referring to the Discrete Fourier Transform (DFT) and its inverse [4]. More specifically, the Fourier transform is an operator $\mathcal{F} : R^{N \times M} \rightarrow C^{N \times M}$ such that, given a

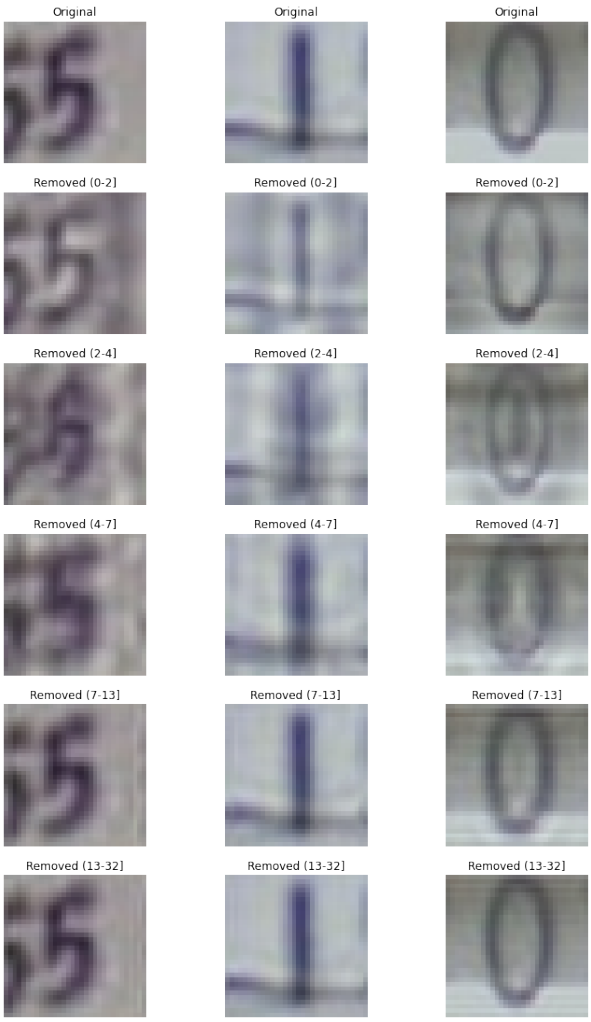

Figure 2. Example of distorted SVHN images according to our model. Notice how the less clear edges on rows 2 and 3 confuse even the human eye of the class of the digit

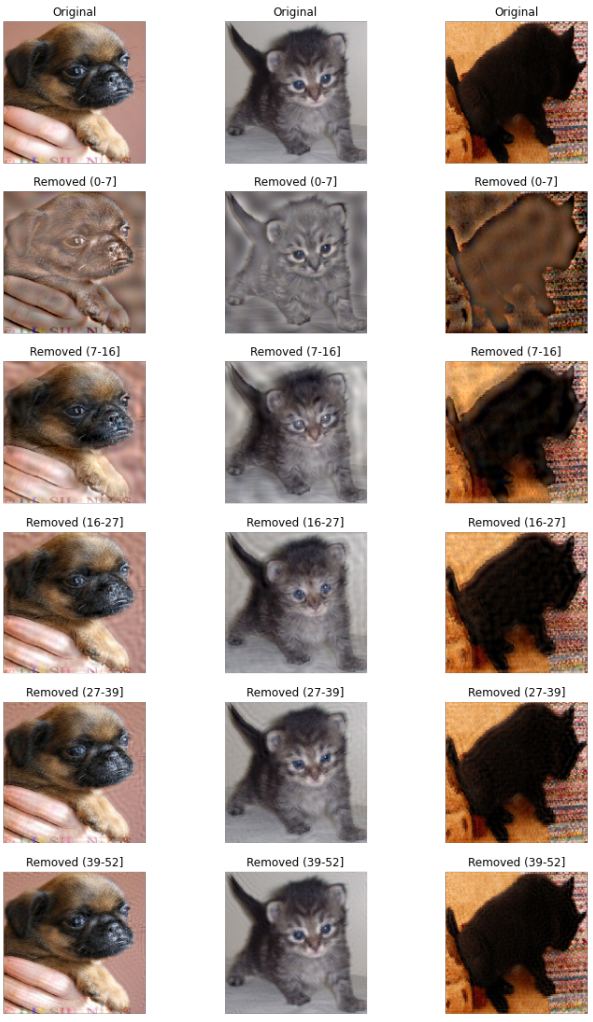

Figure 3. Example of distorted ImageNet images according to our model. Only five intervals amounting each 10% of total energy are shown for brevity. Notice how the effect is barely noticeable by the fifth interval, and how the fur texture is impoverished on the third and fourth intervals

matrix X, results in a complex-valued matrix Y:

$$Y[k,l] = \frac{1}{N*M} \sum_{p=0}^{N-1} \sum_{q=0}^{M-1} X[p,q] e^{-2\pi i (\frac{kp}{N} + \frac{lq}{M})} \quad (1)$$

For each (k,l) pair representing a frequency, the magnitude of the complex coefficient of that frequency is called the energy contributed by $Y[k,l]$ [4].

The DFT's resulting matrices are often shifted to leave the zeroth frequency ($Y[0,0]$) at the center. In this sense, the "distance," "distance from the center," or "size" of a frequency (k,l) is just the norm of the pair. The "height" of a frequency, in the sense of low and high frequencies, also refers to that.

3.1. Assessing Frequency Importance

The DFT provides a simple way to test whether a frequency is important for a model to classify an image. Given a frequency (k,l) and an image X we can construct a frequency mask M defined as:

$$M[p,q] = \begin{cases} 0, & \text{if } (p,q) = (k,l) \\ 1, & \text{otherwise} \end{cases} \quad (2)$$

This frequency mask can then be piece-wise multiplied by the Fourier transform of the image X, yielding the Fourier representation of an image without the frequency (k,l), X'. This representation can then be turned into a pixel representation of that image using the inverse DFT. To put shortly:

Figure 4. Example of distorted VGGFaces2 images according to our model. Only five intervals amounting each 10% of total energy are shown for brevity

$$X' = \mathcal{F}^{-1}C(\mathcal{F}(X) \times M) \tag{3}$$

By comparing a model's prediction on X to X', we can test if a specific frequency was important in classifying the image. If the prediction changes, then this constitutes evidence that the frequency (k, l) was important for the model's decision.

The test can be repeated throughout an entire dataset of images for a more statistically relevant test. We aggregate the information by calculating the difference in accuracy achieved by a model trained on original images tested on both natural and distorted versions of a test dataset. The *estimated importance* of a frequency is the deviance of the distorted version performance to the baseline performance. This nomenclature is reminiscent of feature importance procedures such as Mean Decrease Accuracy, used on tree-based classifiers and other traditional Machine Learning models[11].

3.2. Energy Distribution Model

The frequency importance test can be made with sets of frequencies rather than individual ones, as an individual frequency may cause an insignificant effect. For large images, on the other hand, it may be intractable to test each frequency. Since we are interested in studying the existence of a high or low-frequency bias in CNNs, we group neighboring frequencies in discs according to their distance to the zeroeth frequency. We chose to divide the frequency spectrum in bands, or frequency discs, with each disc represented by two radii r_1 and r_2, containing all the frequencies with L_1 distance greater or equal than r_1 but strictly lesser than r_2. We use the L1-norm rather than the L2-norm as it is more suit for calculating distances in discrete spaces.

To define the radii for the different discs, we refer to the aforementioned concept of energy carried by each frequency. In a sense, the amount of energy each frequency has is related to the amount of information it contains, so we consider it fair to divide the frequency spectrum into bands with the same amount of energy. We name the collection of integer-valued radii $r_1, r_2, ...r_n, r_n \in \mathcal{R}_n$ an energy distribution model, where the frequency band $[r_i, r_{i+1}), i \in [1, n-1]$ represents $\frac{1}{n}$ of the total energy.

To standardize calculations, we resize all images within a dataset to standard image size, so they always have the same amount of calculable frequencies. We also calculate the energy distribution models using the average energy distribution across all images within a test dataset instead of individually per image. This strategy allows the frequency bands to vary between different datasets of different subject matters and remain comparable within each experiment.

We find that this partition of the frequency space is appropriate for several reasons. It is of simple and straightforward interpretation, as it helps us divide the frequency spectrum from low to high frequencies in a one-dimensional fashion. This partitioning approach also allows us to define high and low-frequencies methodically.

The zeroth frequency represents the average intensity of the pixels of the channel. It has a disproportionate and qualitatively different meaning, and its removal causes severe distortion in the image. Therefore it is never included in the calculus of energy and partitioning of the frequency space. Figures 1, 2, 3, and 4 show examples of the images generated by the proposed method.

3.3. Robust and Non-Robust Features

Besides analyzing from our original point of view, in which energy should be compared with importance, another reasonable assumption would be that accuracy loss should be correlated with the amount of distortion introduced by eliminating each frequency disk. From this point of view, the *excess* of performance loss may constitute evidence for a frequency bias. This concept is related to Ilyas et al.[14]'s theoretical framework for studying robust and non-robust features. They develop a toy model in which non-robustness arises from a misalignment of the metric induced by the features with the metric used by adversarial perturbations. Applying this to our case, we study the ratio of the importance of each frequency band, as measured by our method, to the distortion introduced by removing it, as measured by the average L_2 metric (or mean-squared error, MSE) of distorted images with relation to the originals. The highest this ratio is for a frequency band, the more an adversary could exploit it to achieve a high fool ratio while maintaining a low perturbation score.

4. Experimental Setup

In order to study the effect on frequency preference produced by different data, in which discriminative features eventually lie on different parts of the spectrum, we experiment with various datasets, two of general object detection and classification (CIFAR10[18], ImageNet[2]), one of face recognition (VGGFace2[5]) and one of in-the-wild digit recognition (SVHN [21]). We train three distinct network architecture families on each dataset, VGG[22], ResNet [12] and DenseNet[13] to observe the effect of architecture on frequency bias.

Besides this general scenario, we are also interested in two other variables on frequency bias: depth of networks and pre-processing normalization. On the VGGFace2 dataset, we train and compare two versions of each architecture family of different depth. By normalization, we understand the act of subtracting from each sample its mean intensity and dividing it by its standard deviation before passing it to the neural network. To isolate architecture effects, we test the DenseNet architecture on all datasets trained with and without normalization. We divided all values between 0 and 255 by 255 whenever pixel intensities were in that range as an extra pre-processing step for the non-normalized scenarios.

4.1. Datasets

CIFAR10 is a traditional object classification dataset. It consists of 60.000 images with an already standardized 32x32 image size, divided into ten classes. It provides a train/test split of 50.000 images for training and 10.000 images for testing. We used the dataset precisely as provided by the Keras Deep-Learning library [7].

The Street View House Numbers (SVHN) [21] is an in-the-wild digit recognition dataset used for object recognition and object classification. Original images vary in image size and are provided along with bounding boxes for digits, intended for training and evaluation of object recognition. However, the dataset is also available in a cropped format, with each image resized to 32x32, intended for image classification. Collectively, the images have 73.257 digits for training, 26.032 digits for testing, and 531.131 additional examples, according to the official website [21]. We use only the cropped version training and testing sets, as they are provided in the Tensorflow Deep Learning Library [1].

The ImageNet Large Scale Visual Recognition Challenge (ILSVRC or ImageNet, for short)[2] is an annual challenge for object detection and classification with 1.000 classes. Training and validation images are provided with varying image sizes and bounding boxes for points of interest. We used the *Restricted* scenario suggested by Ilyas et al. [14], in which several classes are grouped into nine superclasses of animals. Using the ILSVRC 2017 version, this scenario includes 112.365 training images and 10.150 validation images. We crop the bounding boxes and resize all images to 160x160.

Finally, the VGGFace2 [5] is a Face Recognition dataset. It contains over 3 million images of 9.131 different identities, 8.631 included in the train set, and 500 in the test set. As a dataset of recognition and not classification, the sets of identities on the test and train set are disjoint. We artificially created a train/test classification split using the original training set. We used 5% of the data for the new test set in a stratified manner to ensure fair class representation.

We additionally pre-process VGGFace2 images by cropping the facial images using the bounding boxes provided with the dataset. We amplify the bounding boxes by 20 percent and scale all images to 160x160, interpolating and cropping when necessary but always retaining the original aspect ratio.

The energy distribution model was calculated exclusively on the test set of each dataset. For VGGFace2 and ImageNet, we divided the frequency spectrum into ten discs with 10% of the energy each. For CIFAR10 and SVHN, the small image size made it difficult to divide the calculated frequencies fairly, so we chose to use five discs with 20% of the energy each instead.

4.2. Networks and Training

We chose three different network architecture families that represent the recent evolution of CNNs for Computer Vision. We refer to their respective papers for a more in-depth explanation of their differences but highlight their essential aspects. The VGG network[22] achieved success by effectively building and training much deeper networks than the state-of-the-art. The ResNet[12] introduced residual connections, in which each layer would learn a residual to be added to the input rather than transforming it freely. This strategy proved to be a much more effective way of training CNNs. Finally, the DenseNet[13] built on ResNets by adding Dense blocks, in which every layer received as input the feature maps from each layer preceding it, improving gradient flow along with the network.

For the two datasets with larger image sizes (ImageNet and VGGFace2), the network implementations we used were the ones provided by the Keras Deep-Learning Library[7]. For the CIFAR10 and SVHN cases, we implemented the specific changes described in the ResNet and DenseNet papers to tailor these networks to datasets of smaller sizes. The VGG paper had no experiments in CIFAR10 or similar datasets, but we found that the network was able to perform well on them nevertheless.

All models are trained using standard SGD with 10^-2 learning rate and 0.9 momentum, with different training duration and learning rate schedules depending on each dataset but standardized across networks. Regular data augmentation procedures are applied (random shifts, rotation, zoom, and horizontal flip, except on the SVHN case, where we do not flip images). All image data is normalized unless specified.

5. Results and Discussion

We present two visualizations for the main results. Figure 5 shows the test accuracy for each model on all degradations, along with the baseline accuracy of each model. Figure 6 shows the decrease in accuracy with relation to each model alongside the amount of distortion introduced by each filtering step and the proportion of accuracy decrease to distortion. Figure 7 uses both visualizations for our comparison on different network depth. Our main takeaways from the data are:

- Results change radically in shape and scale across datasets.
- In low-res datasets, higher frequencies tend to be disproportionately important, but the effect is less prominent on high-res ones.

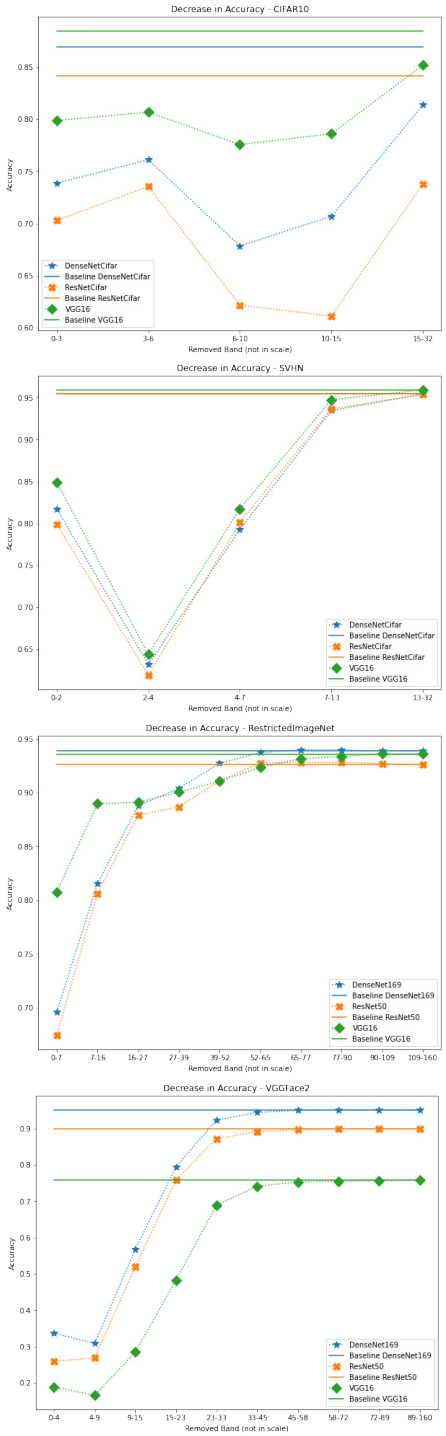

Figure 5. Accuracy vs removed frequencies (frequencies not to scale)

- Comparing accuracy decrease with MSE, mid to higher frequencies universally had a higher ratio.
- Network architectures may produce a small difference in scale, but not in the shape of the effect.

Figure 6. Comparison of MSE of degraded images and decrease in performance

- Network depth specifically does not seem to play a role in frequency bias.

Our results seem to reproduce the ones found by Jo and Bengio[17]. On the datasets they studied, we can see that higher frequencies (the second and third on SVHN and the third and fourth on CIFAR10) affected the classifiers more than the lower ones, suggesting some high-frequency bias. However, when we expanded our research to more datasets, we found that this phenomenon is not universal, as lower frequencies tend to be more critical in RestrictedImageNet and on VGGFace2, with the second frequency disc being slightly more critical on VGGFace2 in some cases. We can also see that the effect on different models is more of scale than of curve's shape. This fact suggests that the frequency bias is not related to the peculiarities of CNN architectures, either related to the universal properties of CNNs or data patterns. Image size plays a significant role in the frequency bias, as the curve varies most between the lower resolution (CIFAR10 and SVHN, with 32x32 images) and higher resolution (ImageNet and VGGFace2, which were scaled to 160x160). The difference between ImageNet and VGGFace2 can also be attributed to differences in the datasets' objectives, as discriminative facial features would lie on a higher frequency mode. This hypothesis will be the subject of further work.

From the point of view of robust features, Fig. 6 shows a more precise pattern, in which the discs with more decrease in accuracy per MSE are always in higher frequency modes. The reason is either that these discs were the ones with higher importance (CIFAR10, SVHN) or because the MSE distortion decreased way faster than the accuracy loss (ImageNet, VGGFace2). That points out that networks may be learning non-robust features in higher frequency modes, which can, in turn, be exploited in an adversarial setting, as also suggested by Wang et al. [24]. From either point of view, there is some evidence for a frequency bias. Interestingly enough, in both cases, it seems it would be more appropriate to name it a "mid-frequency" bias than a "high-frequency" one, a result similar to the one of Tsuzuku et al. [23].

We observe almost no significant patterns when comparing across network architectures. Our experiment on ImageNet seems to corroborate Geirhos et al. [9], which found that VGG-like networks were more prone to classifying ImageNet based on texture rather than shape, and Wang et al. [24], which found them more prone to learning from high-frequency components. Figure 6 shows how the decrease per MSE ratio for the VGG network lingers on and is the slowest to recede not only on the ImageNet but on the VGGFace2 case. However, this pattern was not found on the two low-res experiments, so it may not be a universal attribute of VGG-like networks.

Figure 7 shows that on our more variable-specific experiments, network depth has minimal effect on the frequency bias. Deeper models seem to attain a better accuracy at the cost of some loss in robustness, especially in the ResNet and DenseNet cases. However, looking through the robust features lens, the deeper networks seem to be less prone to high-frequency bias, which is somewhat surprising. Our experiments with normalization were not conclusive, with the results varying more across datasets and little by our confusing variable, not yielding any significant pattern.

6. Conclusion and Future Developments

We studied the common hypothesis that CNNs are prone to over-rely on imperceptible high-frequency patterns. We developed a method that allowed us to study

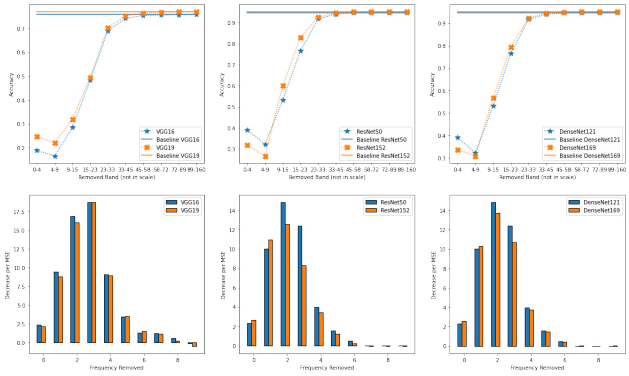

Figure 7. Effect of depth on different models trained on VGGFace2

how CNNs respond to different frequencies on different conditions. While our method has not yielded any quantitative metric, it is an improvement to the current state-of-the-art of research as it divides the frequency spectrum systematically using reasonable assumptions instead of relying on the researcher's discretion.

We found no clear-cut evidence for or against a high-frequency bias. However, we found some evidence that indicates CNNs tend to value more mid to high frequencies. This phenomenon also varied much more across datasets than by any other variable we studied, indicating that this may be more of a data phenomenon than a model phenomenon. We find it thus, improbable that the high-frequency bias hypothesis can explain the entirety of CNN's brittleness or capability of generalization. Our model could be applied as-is to research the effects of various other components of modern CNNs, such as Batch Normalization[15] or Adversarial Training[20]. For this reason, we provide open source code along with this paper.

There is also plenty of room for improvement in our model. Other strategies to divide the frequency spectrum could prove more informative, such as dividing by equal amounts of distortion introduced. Our analysis could also be complemented by estimating how much useful information is contained on each frequency disk, perhaps by training models exclusively on each disk. We are also interested in understanding how methods for training robust networks such as pre-training [6, 16] or architecture optimization [8] would affect those observations.

References

[1] Martín Abadi, Ashish Agarwal, Paul Barham, Eugene Brevdo, Zhifeng Chen, Craig Citro, Greg S. Corrado, Andy Davis, Jeffrey Dean, Matthieu Devin, Sanjay Ghemawat, Ian Goodfellow, Andrew Harp, Geoffrey Irving, Michael Isard, Yangqing Jia, Rafal Jozefowicz, Lukasz Kaiser, Manjunath Kudlur, Josh Levenberg, Dandelion Mané, Rajat Monga, Sherry Moore, Derek Murray, Chris Olah, Mike Schuster, Jonathon Shlens, Benoit Steiner, Ilya Sutskever, Kunal Talwar, Paul Tucker, Vincent Vanhoucke, Vijay Vasudevan, Fernanda Viégas, Oriol Vinyals, Pete Warden, Martin Wattenberg, Martin Wicke, Yuan Yu, and Xiaoqiang Zheng. TensorFlow: Large-scale machine learning on heterogeneous systems, 2015. Software available from tensorflow.org.

[2] Alex Berg, Jia Deng, and L Fei-Fei. Large scale visual recognition challenge (ilsvrc), 2010. *URL http://www. image-net. org/challenges/LSVRC*, 3, 2010.

[3] Wieland Brendel and Matthias Bethge. Approximating cnns with bag-of-local-features models works surprisingly well on imagenet. *arXiv preprint arXiv:1904.00760*, 2019.

[4] S Allen Broughton and Kurt Bryan. Discrete fourier analysis and wavelets. In *Applications to signal and image processing*. Wiley Online Library, 2009.

[5] Qiong Cao, Li Shen, Weidi Xie, Omkar M Parkhi, and Andrew Zisserman. Vggface2: A dataset for recognising faces across pose and age. In *2018 13th IEEE International Conference on Automatic Face & Gesture Recognition (FG 2018)*, pages 67–74. IEEE, 2018.

[6] Tianlong Chen, Sijia Liu, Shiyu Chang, Yu Cheng, Lisa Amini, and Zhangyang Wang. Adversarial robustness: From self-supervised pre-training to fine-tuning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 699–708, 2020.

[7] François Chollet et al. Keras. `https://keras.io`, 2015.

[8] Minjing Dong, Yanxi Li, Yunhe Wang, and Chang Xu. Adversarially robust neural architectures. *arXiv preprint arXiv:2009.00902*, 2020.

[9] Robert Geirhos, Patricia Rubisch, Claudio Michaelis, Matthias Bethge, Felix A Wichmann, and Wieland Brendel. Imagenet-trained cnns are biased towards texture; increasing shape bias improves accuracy and robustness. *arXiv preprint arXiv:1811.12231*, 2018.

[10] Ian J Goodfellow, Jonathon Shlens, and Christian Szegedy. Explaining and harnessing adversarial examples. *arXiv preprint arXiv:1412.6572*, 2014.

[11] Hong Han, Xiaoling Guo, and Hua Yu. Variable selection using mean decrease accuracy and mean

decrease gini based on random forest. In *2016 7th ieee international conference on software engineering and service science (icsess)*, pages 219–224. IEEE, 2016.

[12] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016.

[13] Gao Huang, Zhuang Liu, Laurens Van Der Maaten, and Kilian Q Weinberger. Densely connected convolutional networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 4700–4708, 2017.

[14] Andrew Ilyas, Shibani Santurkar, Dimitris Tsipras, Logan Engstrom, Brandon Tran, and Aleksander Madry. Adversarial examples are not bugs, they are features. In *Advances in Neural Information Processing Systems*, pages 125–136, 2019.

[15] Sergey Ioffe and Christian Szegedy. Batch normalization: Accelerating deep network training by reducing internal covariate shift. *arXiv preprint arXiv:1502.03167*, 2015.

[16] Ziyu Jiang, Tianlong Chen, Ting Chen, and Zhangyang Wang. Robust pre-training by adversarial contrastive learning. *arXiv preprint arXiv:2010.13337*, 2020.

[17] Jason Jo and Yoshua Bengio. Measuring the tendency of cnns to learn surface statistical regularities. *arXiv preprint arXiv:1711.11561*, 2017.

[18] Alex Krizhevsky, Geoffrey Hinton, et al. Learning multiple layers of features from tiny images. 2009.

[19] Jake Lee, Junfeng Yang, and Zhangyang Wang. What does cnn shift invariance look like? a visualization study. *arXiv preprint arXiv:2011.04127*, 2020.

[20] Aleksander Madry, Aleksandar Makelov, Ludwig Schmidt, Dimitris Tsipras, and Adrian Vladu. Towards deep learning models resistant to adversarial attacks. *arXiv preprint arXiv:1706.06083*, 2017.

[21] Yuval Netzer, Tao Wang, Adam Coates, Alessandro Bissacco, Bo Wu, and Andrew Y Ng. Reading digits in natural images with unsupervised feature learning. 2011.

[22] Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*, 2014.

[23] Yusuke Tsuzuku and Issei Sato. On the structural sensitivity of deep convolutional networks to the directions of fourier basis functions. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 51–60, 2019.

[24] Haohan Wang, Xindi Wu, Zeyi Huang, and Eric P Xing. High-frequency component helps explain the generalization of convolutional neural networks. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 8684–8694, 2020.

[25] Dong Yin, Raphael Gontijo Lopes, Jon Shlens, Ekin Dogus Cubuk, and Justin Gilmer. A fourier perspective on model robustness in computer vision. In *Advances in Neural Information Processing Systems*, pages 13255–13265, 2019.

[26] Chiyuan Zhang, Samy Bengio, Moritz Hardt, Benjamin Recht, and Oriol Vinyals. Understanding deep learning requires rethinking generalization. *arXiv preprint arXiv:1611.03530*, 2016.